

AD-A228 442

RADC-TR-90-205
Final Technical Report
September 1990



ALOPEX OPTIMIZATION ALGORITHM

Syracuse University

E. Harth, T. Kalogeropoulos, W. Liu, A.U. Joshi



APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

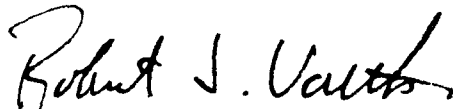
This effort was funded totally by the Laboratory Director's Fund.

Rome Air Development Center
Air Force Systems Command
Griffiss Air Force Base, NY 13441-5700

This report has been reviewed by the RADC Public Affairs Division (PA) and is releasable to the National Technical Information Services (NTIS) At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-90-205 has been reviewed and is approved for publication.

APPROVED:



ROBERT J. VAETH
Project Engineer

APPROVED:



RAYMOND P. URTZ, JR.
Technical Director
Directorate of Command & Control

FOR THE COMMANDER:



IGOR G. PLONISCH
Directorate of Plans & Programs

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (COTC) Griffiss AFB NY 13441-5700. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OPM No. 0704-0188

Please reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Information and Regulatory Affairs, Office of Management and Budget, Washington, DC 20503.

1. AGENCY USE ONLY (Leave Blank)		2. REPORT DATE September 1990	3. REPORT TYPE AND DATES COVERED Final Jan 89 - Dec 89	
4. TITLE AND SUBTITLE ALOPEX OPTIMIZATION ALGORITHM			5. FUNDING NUMBERS C - F30602-88-D-0027 PE - 61101F PR - LDFFP TA - 09 WU - P9	
6. AUTHOR(S) E. Harth, T. Kalogeropoulos, W. Liu, A.U. Joshi				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Syracuse University Office of Sponsored Programs Skytop Office Bldg - Skytop Road Syracuse NY 13244-5300			8. PERFORMING ORGANIZATION REPORT NUMBER N/A	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Systems Command Rome Air Development Center/COTC Andrews AFB MD 20331 Griffiss AFB NY 13441-5700			10. SPONSORING/MONITORING AGENCY REPORT NUMBER RADC-TR-90-205	
11. SUPPLEMENTARY NOTES RADC Project Engineer: Robert J. Vaeth/COTC/(315) 330-2925 This effort was funded toally by the Laboratory Director's Fund.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This report provides results of the investigation and application of the Alopex Algorithm to optimization problems. The Alopex Algorithm is a stochastic multiparameter optimization procedure. This algorithm addresses the two optimization problems of resource allocation and pattern recognition.				
14. SUBJECT TERMS Algorithm, Optimization, Stochastic			15. NUMBER OF PAGES 54	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

INTRODUCTION

The assigned tasks consist of two parts: (1) Recognition of 3-D objects given a 2-D projection. (2) Optimal resource allocation. Both assignments involved application of the optimization algorithm Alopex which was developed by the principal investigators.

Our investigation on these assignments are described in section I and II. We investigated 3-D objects with a view toward identification of airplanes. A system capable of rapid identification of airplanes can be built based on our findings. For the second assignment we investigated the problem of assigning in an optimal way a number of guns to targets. Alopex has been found to converge rapidly to optimal assignments. We also compared the performance of the Alopex algorithm with the widely used method of simulated annealing. Alopex was found to converge considerably faster in all examples tried.



Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input checked="" type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

I. 3-D PATTERN RECOGNITION

1. Introduction

The identification of three-dimensional objects from their two-dimensional images suffers many difficulties. In addition to the loss of one dimension, magnification and orientation of the imaged object are usually unknown. The method of identifying three-dimensional objects has been investigated by a number of researchers.

In order to retain the translation, rotation, size and shape information, Dudani et al^[8] calculated up to fourteen moments as features to represent the image, while Wallace et al^[20] used a complex representation of the boundary curves to obtain Fourier coefficients as the set of descriptors. In both reports, libraries containing the representations of candidate objects had to be constructed.

Chien et al^[6] proposed a method which needs multiple views of the object, represented by structures called quadrees and octrees. Banhu,^[2] Wang^[21] and Ben-Arie^{[3][4][5]} used other sets of features to match pictures with objects.

In our method the input pattern (picture) is represented by a two-dimensional matrix whose elements (zeroes and ones) are pixels. The description of the three-dimensional objects (templates) consists of three-dimensional arrays of zeroes and ones specifying the boundaries. Comparisons are made between picture and templates by forming orthogonal projections of the templates using different Euler angles. Our method, unlike the others, requires less storage to describe the objects but defines a dynamic process by which the simple stored template information is transformed to match the picture.

2. Method

In our studies a *picture* (2-D input) is identified with one of a set of known objects (*templates*). Binary arrays are used, with object represented by ones and background by zeroes. The templates are arrays of 64x64x64 elements. The pictures are matrices of 64x64 elements.

2.1 Rotation of the Templates

Since a projection has to be obtained before comparison can be made, three angles need to be chosen. We are using three Euler angles to represent the rotation of three-dimensional objects. There are many choices for the three Euler angles. For example, in a right-handed coordinate system, there are twelve sets of Euler angles.^{[16][17]} For our purpose, the following set is chosen. Let the coordinate system be $Oxyz$ before rotation and $OXYZ$ after rotation. Let OA be the line of intersection between the Oxy and OXY planes. Define as θ_1 the angle between Oz and OZ axes, θ_2 that between OA and Ox and θ_3 the angle between OA and OX . The ranges of these angles are:

$$0^\circ \leq \theta_1 < 180^\circ, 0^\circ \leq \theta_2 < 360^\circ, 0^\circ \leq \theta_3 < 360^\circ.$$

The templates are rotated using these angles and projected on the Oxy plane. The disparity between a template projection and the picture is represented by a *cost function*, which we wish to optimize.

2.2 The Alopex Algorithm

The Alopex algorithm was proposed for solving optimization problems.^{[9][10][19]} The optimization procedure is stochastic and iterative. In every iteration, all variables that determine the cost function change by small increments, and the cost function is computed. The change of a variable depends stochastically on the change of the cost function and the change of the variable as calculated from the prior two iterations. Two parameters control the optimization process: the *probability* or *stochasticity* p and the *step size* δ . The Alopex algorithm used here is defined below.

Let $F(x_1, \dots, x_n)$ be the *cost function* and x_1, \dots, x_n the variables to be adjusted. The change in the i -th variable at the n -th iteration is given by

$$x_i^{(n)} = x_i^{(n-1)} + \delta_i^{(n)} \quad (1.1)$$

where the increments $\delta_i^{(n)}$ are given by

$$\delta_i^{(n)} = \begin{cases} P_i^{(n)} \delta & \text{with probability } p \\ -P_i^{(n)} \delta & \text{with probability } (1 - p) \end{cases} \quad (1.2)$$

where

$$P_i^{(n)} = \begin{cases} +1 & \text{if } m * [x_i^{(n-1)} - x_i^{(n-2)}] * [F^{(n-1)} - F^{(n-2)}] > 0 \\ -1 & \text{if } m * [x_i^{(n-1)} - x_i^{(n-2)}] * [F^{(n-1)} - F^{(n-2)}] < 0 \end{cases} \quad (1.3)$$

The value of m is $+1$ for maximization, -1 for minimization. Similar algorithms have been employed by us successfully in many optimization problems.^[15]

3. Examples and Results

We explored several types of cost functions such as those which match boundaries, or areas, or a combination of both. We also used different optimization

procedures by considering cost functions that are to be minimized or maximized. We found that minimization of the following cost function is effective:

$$F^j = \sum_i (I_i^k - T_i^j)^2 \quad (1.4)$$

where j is the template number, k the particular template from which the picture is formed, i is the pixel and I_i^k, T_i^j are zeroes or ones representing the input (picture) and orthogonal projection of the template j respectively. The input is defined by the three Euler angles $\Theta_1^k, \Theta_2^k, \Theta_3^k$. The template projection is defined by the Euler angles $\theta_1^j, \theta_2^j, \theta_3^j$. The general problem is to obtain the global minimum of F^j by varying the angles $\theta_1^j, \theta_2^j, \theta_3^j$.

In employing the Alopex algorithm, many different step sizes have been tried, and it was found that they affect the speed of convergence. A δ of about one degree is appropriate for good convergence. If δ is smaller more iterations are needed to reach the optimum, while larger δ 's produce larger fluctuations in the cost function.

The probability p plays also a very important role in the process. It is necessary to keep p less than 1 to prevent trapping of the process in local extrema and larger than 0.5 in order to drive the process towards optimization. A good operational value is around 0.75.

In understanding this general problem the following classes of problems have been investigated.

3.1 One 3-D Template

We chose a single template in the shape of the letter L. The input is an arbitrary orthogonal projection of the template. In this study we test the convergence of the angles $\tilde{\theta}$ to $\tilde{\Theta}$ as a function of step δ and probability p . We illustrate the convergence of the process by presenting results shown in Figs. 1-3. The step size and probability

were kept at 1° and 0.75 respectively. In these examples, which are typical of most runs, the optimum was reached within a few hundred iterations. At the optimum, the variables θ_1, θ_2 and θ_3 reached the values of the input parameters Θ_1, Θ_2 and Θ_3 .

In Fig. 1 the process started with initial angles $\theta_1 = 60^\circ$, $\theta_2 = 270^\circ$, and $\theta_3 = 290^\circ$ while the template input angles were $\Theta_1 = 110^\circ$, $\Theta_2 = 230^\circ$ and $\Theta_3 = 335^\circ$. It is seen that the cost decreased to a minimum in about 200 iterations, and the corresponding input angles were found. Notice that the initial angles differed from the input angles by less than 45° .

In Fig. 2 the differences between input and initial angles were larger. The cost function reached a local minimum in about 150 iterations and was trapped for a while and then reached the global minimum. This shows that the cost function, in order to reach the global minimum, had to overcome some local minima. If this difference is further increased, the cost function may not reach the global minimum with the chosen step size and probability. There are many local minima in this three-dimensional pattern recognition cost function.

Fig. 3 shows the result of an annealing schedule, in which the step size was reduced from 1° to 0.2° after 200 iterations, otherwise the conditions were the same as those in Fig. 1.

3.2 Five 2-D Templates

In order to further investigate the problem of convergence with highly disparate initial and input angles the simpler problem of 2-D template identification was considered. Five templates were formed by crossing two rectangles. The templates are similar in appearance differing only by 50 to 100 pixels (Fig. 4). In Fig. 5 the costs, F_j , for the five templates are shown as functions of iteration number. The

input was derived from template **V** using an input angle $\Theta = 190^\circ$. The process was initialized with angles $180^\circ, 150^\circ, 220^\circ, 230^\circ$ and 265° for the five templates respectively. Note that template **V** had the largest initial disparity which is reflected by the largest initial cost function (red trace in Fig. 5). As seen, the cost function for template **V** reached the lowest value, and thus made the correct identification. Many simulations have been done with different input and initial conditions. We found that the input can be recognized as long as the difference between input and initial angles is within 90° because of a deep local minimum around input $\theta = \Theta + 180^\circ$.

3.3 3-D Object Identification

In the above cases we studied identification from 2-D templates and tests of convergence for a single 3-D template. Now we present simulation results of object identification using several 3-D templates.

Two 3-D templates

Two templates are used here, which are the 2-D patterns used above (Fig. 4) with a common thickness added as a third dimension. The parameters δ and p in Alopex were the same as before. The rotation variables of the template were independent of one another. We found that correct recognition and assignment of angles were achieved if the input and initial angles were within $\pm 45^\circ$.

Five 3-D object identification

In this study, the templates used are the five 3-D objects resembling airplanes whose x - y projections are shown in Fig. 6. Along the z -axis different thicknesses were used. The templates differ between 30 and 200 pixels. In the Alopex algorithm,

we start with arbitrary Euler angles ($\vec{\theta}$) defining the template projections. The cost function used is given by equation (1.4). We have found that Alopex always converges but not necessarily to the global minimum which defines the correct identification. As it was found above in the case of two 3-D templates, if the initial value of the variables were within 45° , Alopex made the correct identification. By running Alopex a number of times and starting with random initial values for the Euler angles, the global minimum is always found as illustrated by the following two examples.

Example 1:

The two parameters used for controlling the Alopex process in Eq. (1.2) are $\delta = 1$ and $p = 0.78$. Table 1 shows the results of four optimization runs. In each run the input is a projection of the template 2 with input angles listed. Each run consists of six independent trials in which random initial orientations are chosen for each of the five templates. Each trial was terminated after 700 iterations. The Table lists for each trial the lowest cost function, the winning template and the iteration at which the minimum was reached. As seen in each run the correct template was identified.

Example 2:

In this example, the same parameters δ and p were used. The inputs are from template 4 with different set of angles. Again, it can be seen in Table 2 that the object was correctly identified.

In addition to the cost function defined by Eq. (1.4) we have investigated the effectiveness of other cost functions such as moments, boundaries and others. Thus far the one defined by equation (1.4) gives the best results.

4. Conclusions

From our extensive computer simulations on 3-D pattern recognition, given a 2-D projection as input, we conclude:

- (a) The Alopex optimization converges to the global minimum if the initial orientation of the template is within 45° of the input values.
- (b) In the most general case in which the input orientation is totally unknown, the correct identification can always be achieved by performing a small number of trials with randomly chosen initial orientations.

II. OPTIMAL RESOURCE ALLOCATION

1. Introduction

Resource allocation is a well known *NP-complete* class of problems. No polynomial time solution is known. However, stochastic methods have been classically applied, some of which have given good results but with long, impractical run-times^[18] for large problems. ALOPEX being *finely* parallel, can achieve optimization, within reasonable time limits. Further substantial improvement is possible, if the program is implemented on a parallel machine like the Connection Machine 2. What remains to be done is to investigate methods of controlling the dynamics of the problem.

It is the objective of this research to obtain the best set of *assignments* for different *facilities* to perform a given set of *tasks*. We are given the efficiencies with which each facility can carry out each task, and the *interactions*, i.e., the extend to which one task interferes with or enhances the efficiency of a facility to carry out a different task.

To make the problem more specific, we take the *facilities* to be *guns* and the *tasks* to be *hitting targets*. Other examples of the same formulation of the resource assignment problem easily come to mind, such as:

- (1) Assignment of proper jobs to human resources.
- (2) Distribution of resources to various demand locations in a cost effective way.
- (3) Best use of facilities and personnel in disaster relief.

2. Specific Formulation

Our approach uses the Alopex method of optimization, and thus explores the parallelism inherent in the resource allocation problem.

Given a set of guns and a set of targets, we make initially random assignments. These assignments are changed incrementally and iteratively, using the Alopex algorithm^[9] and a scalar *cost function* constructed from the expected *success rate* for the various targets.

We consider a specific formulation of the problem as defined by the following quantities and relations. A number of practical problem can be represented within this formalism, and the effectiveness of the Alopex optimization can be evaluated.

We define:

ϵ_{ij} : *Efficiency* of gun i to hit target j .

α_{ij} : *Assignment* of gun i to fire at target j , where

$$\sum_j \alpha_{ij} = 1 \text{ and } 0 \leq \alpha_{ij} \leq 1.0 \quad (2.1)$$

σ_{ijk} : *Mutual Enhancement coefficient* of gun i to hit target j , if gun i is also assigned to target k .

We assume the performance of any gun i to be limited by the normalization condition given by Eq. (2.1). The resource allocation problem is, therefore, defined as the proper adjustment of α_{ij} which denote the *assignments* of guns to targets with the aim of maximizing the probability of hits.

With our formulation, the probability of gun i hitting target j is:

$$P_{ij} = \alpha_{ij} \cdot (\epsilon_{ij} + \sum_k (\alpha_{ik} \cdot \sigma_{ijk})) \quad (2.2)$$

Equation (2.2) expresses our assumption that the efficiency ϵ_{ij} is enhanced (or reduced) by an amount determined by the assignments of the gun i to other targets k and the interaction coefficients σ_{ijk} .

The probability of a target j being hit is therefore:

$$P_j = 1 - \prod_i (1 - P_{ij}) \quad (2.3)$$

We have constructed a *cost function*,

$$F = \sum_j (1 - P_j)^2 \quad (2.4)$$

which is to be minimized by varying α_{ij} .

3. Experiments and Results

In a series of computer simulations we have examined the performance of the Alopex algorithms when applied to a variety of problems in which we have assumed different numbers n of facilities (guns) to be assigned to the performance of m tasks (hitting targets). Different distributions of efficiencies ϵ_{ij} and interactions σ_{ijk} were assumed.

Some results of these preliminary investigations are presented. Two parameters determine the dynamics of the optimization process. These are the step size δ by which the assignments α_{ij} are changed in each iteration, and the probability p that enters into the Alopex algorithms^[9]. In the following simulation runs we present the convergence properties of the algorithm. In the following simulation runs we present the convergence properties of the algorithm. The α_{ij} were initially assigned random values between 0 and 1, subject to the normalization condition Eq. (2.1).

Example 1:

We assume that 2 guns are available to fire at 3 targets. The efficiencies ϵ_{ij} are given in Table 3(a). It is seen that with our assumptions gun 1 has a high efficiency for hitting target 2, gun 2 has reasonably high efficiencies for hitting targets 1 and 3, but is poor in hitting target 2. It is intuitively obvious that gun 1 should be

assigned mostly to target 2 and perhaps part of the time to target 3 while gun 2 should divide it's efforts between targets 1 and 3.

In Table 3(b) we show the result of an optimization run in which the step size was $\delta = 0.015$, since the α_{ij} range from 0 to 1.0 (this is 1.5% of the dynamic range) and the probability p in the Alopex algorithm (Eq. 1.1-1.3) was assumed to be $p = 0.75$. In this run all interactions σ_{ij} were taken to be zero. We see from Table 3(b) that our intuitive expectations are confirmed by the assignments generated by Alopex. Figure 7 shows the cost function F as a function of iteration numbers.

Example 2:

In another simulation, we took 4 guns and 4 targets. Table 4(a) shows the efficiencies of guns for each of the targets. Note that, again, the efficiencies suggest an obvious assignment: gun 1 is best for hitting target 2, gun 2 is good only for target 3, gun 3 for target 1 and gun 4 for target 4. When Alopex is applied to this problem, with σ_{ijk} kept at zero, as in the previous example, we obtain the expected results reflected by the high assignments of the guns to the targets they are best in hitting as seen in Table 4(b). Note the high probabilities of hitting the targets.

Figure 8 shows the cost function plotted against iteration numbers for this example. A $\delta = 0.02$ and $p = 0.80$ was used in the Alopex algorithm.

Example 3:

In this example, we again took 4 guns and 4 targets, but the efficiencies were assumed as shown in Table 5(a). The efficiencies do not suggest any obvious assignments, because some of the guns have equal efficiencies in hitting 2 or more targets, and some targets can be hit equally well by 2 or more guns. When Alopex was applied to this problem with $\delta = 0.01$ and $p = 0.75$, we get the assignments for the best cost function in 1000 iterations as shown in Table 5(b). Figure 9 shows the variation of cost function versus iteration number.

Example 4:

Two separate simulations were conducted with the same number of guns and targets and also the same associated efficiencies as in Example 2 except non-zero values for σ_{ijk} were used. In the first simulation we set the mutual enhancement coefficients $\sigma_{ijk} = +0.1$, thus enhancing the efficiencies. This input gave higher probabilities P_j . The cost function shown in Figure 10(a) reached low values within 1000 iterations. In the second simulation we chose negative mutual enhancement coefficients $\sigma_{ijk} = -0.1$, thus reducing the efficiencies. This reduced the probabilities P_j for all targets, compared with the run where $\sigma_{ijk} = 0.0$ (Example 2). Figure 10(b) shows the cost function versus iteration number. Note that the cost remains high.

4. Timing Issues

Figure 11 is a plot of number of parameters α_{ij} to be optimized (with is the product of the number of guns and number of targets) versus the VAX-8800 (ULTRIX OS) cpu time per 1000 iterations. As can be seen, the cpu time is linear to the number of parameters within the range of parameters considered.

We find,

$$t = 1.01 * (\text{No. of parameters}) \text{ sec}/1000 \text{ iterations.}$$

As every parameter is optimized independent of others parameters in every iteration, dependent only on the scalar value F , the computation time would be expected to vary linearly with the *square root* of the number of parameters, if the program is run on a parallel machine like the Connection Machine - 2. This, of course, depends on how the problem is mapped. In fact a properly mapped problem could be solved in a time that depended weakly on the number of parameters, depending on the architecture of the machine.

5. Comparison with Simulated Annealing

We have successfully applied the Alopex algorithm to the resource allocation problem and good assignments have been obtained for a set of guns to hit a set of targets. Our simulations have shown that the Alopex algorithm is applicable to such a problem.

The other stochastic optimization algorithm is the widely used simulated annealing^{[7][11]}. We will compare the performance of Alopex with simulated annealing for the resource allocation problem.

5.1 Simulated Annealing (SA)

This is a heuristic optimization technique. The algorithm, based on the Metropolis^[14] method, is stochastic and iterative. In every iteration the cost function is evaluated after performing random changes in the variables. If the cost function has improved over the preceding iteration, the changes in the variables are kept. Otherwise the changes are accepted with a probability

$$p = \exp(-\Delta F/\beta)$$

where β is an adjustable parameter, analogous to temperature in statistical mechanics. The process of annealing consists of devising a schedule for reducing β as the iteration number increases. Customarily, β is kept constant over a number of iteration L , called the chain length. The process is then described in terms of a Markov chain.

5.2 Numerical Examples

In the problem of guns and targets, the same equations, (2.1)-(2.4), are used for Alopex and SA to optimize the assignments, α_{ij} . Identical initial conditions and step sizes, δ , are used.

In the first example, we consider 2 guns firing at 3 targets. The enhancements σ_{ijk} are taken to be zero and efficiencies ϵ_{ij} are those given in Table 3(a). We used as control parameters $\delta = 0.015$, $p = 0.78$ in Alopex, and $\delta = 0.015$, $L=50$, $\beta_0 = 0.5$ in simulated annealing. The control value β is decreased by 10% after each chain length.

Figure 12(a) shows the evolution of the cost function for Alopex and 12(b) that for SA. It can be seen that the optimal result is obtained by both methods, but Alopex converges in about 220 iterations compared with the 2500 iterations required in SA.

The second example treats 2 guns and 4 targets. The efficiencies are shown in Table 6. Now the chain length is chosen to be $L = 100$. The remaining parameters are the same as in example 1. Fig. 13(a),(b) again show the superior performance of Alopex. Notice that no annealing has been used in the Alopex. Past experience has shown that the Alopex performance can be improved by gradually increasing p (Eq. 1.2) which is analogous to annealing.

6. Conclusions

The problem of the resource allocation has been investigated by applying Alopex to a specific problem. We conclude:

- (a) Alopex could be applied to practical problem of this type.
- (b) The performance of Alopex was superior to SA in the examples investigated.

References

- [1] E. H. L. Aarts and P. J. M. van Laarhoven, "Statistical cooling: a general approach to combinatorial optimization problems," *Philips Journ. Res.* vol. 40, pp. 193-226, 1985.
- [2] B. Banhu, "Representation and shape matching of 3-D objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-6 No.3, 1984.
- [3] Jezekiel Ben-Aroie and A.Z. Mriri, "3-D objects recognition by optimal matching search of multinary relation graphs," *Computer Vision, Graphics and Imaging Processing*, vol.37, No.3, Mar. 1987.
- [4] Jezekiel Ben-Aroie and A.Z. Mriri, "3-D objects recognition by state space search:optimal geometric matching," *IEEE Computer Vision, pattern recognition Conference*, Miami, June 1986.
- [5] Jezekiel Ben-Aroie and A.Z. Mriri, "The area ratio method used for 3-D object recognition," *Proceedings, Machine Intelligence Conference 2*, London, Nov. 1985.
- [6] C.H. Chien and J.K. Aggarwal, "Identification of 3-D objects from multiple silhouettes using quadtrees/octrees," *Computer Vision, Graphics and Imaging Processing*, vol.36, No.2, Nov. 1986.
- [7] Lawrence Davis, *Genetic Algorithm and Simulated annealing*, Morgan Kaufmann Publishers, Inc., Los Altos, California, 1987.
- [8] Sahibsingh A. Dudani, K.J. Breeding and R.B. McGhee, "Aircraft identification by moment invariants," *IEEE Trans. Computers*, vol. C-26, No.1, Jan. 1977.
- [9] E. Harth, T. Kalogeropoulos and A.S. Pandya, "A universal optimization network," *IEEE-EMBS The 10th Annual International Conference Special Symposium Volume*, New Orleans, Nov. 1988.
- [10] E. Harth and E. Tzanakou, "A stochastic method for determining visual receptive fields," *Vision Res.*, vol.14, 1974.

- [11] S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi, "Optimization by Simulated annealing," *Science*, vol. 220, May, 1983.
- [12] Richard H. Leftwich, *The Price System and Resource Allocation*, The Dryden Press, pp. 383-398, 1979.
- [13] Stephen Douglas McRae, *Resource Allocation Alternatives in an Andean Herding System: A Simulation Approach*, Ph.D dissertation, Cornell University, Ithaca, NY, 1979.
- [14] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth and A. H. Teller, E. Teller, "Equation of state calculations by fast computing machines," *J. of Chem. Phys.*, vol. 21, No. 6, pp. 1087-1092, June 1953.
- [15] A.S. Pandya, *A Stochastic Parallel Optimization Algorithm*, Ph.D dissertation, Dept. of Physics, Syracuse Univ., June 1988.
- [16] R.L. Pio, "Euler angle transformation," *IEEE Aerospace and Navigation Electronics*, vol. AC-11, Oct. 1966.
- [17] R.L. Pio, "Symbolic representation of coordinate transformation," *IEEE Aerospace and Navigation Electronics*, 1965, pp. 103.
- [18] Stanley Reiter, *Palgrave's Dictionary of Political Economy*, pp. 107-120, 1988.
- [19] T. Tzanakou, R. Michalak and E. Harth, "The alopex process," *Biol. Cybern.*, 35, pp. 383-398, 1979.
- [20] Timothy P. Wallace and Paul A. Wintz, "An efficient 3-D aircraft recognition algorithm using normalized Fourier descriptors," *Computer Graphics and Imaging Processing*, vol.13, 1980.
- [21] Y.F. Wang, M. Magee and J.K. Aggarwal, "Matching three-dimensional objects using silhouettes," *IEEE Trans. Pattern Anal. Mach. Intell.*, PAMI-6 No.4, 1984.

Figure Captions

- Fig. 1 Cost function versus iteration number for a 3-D template (L-shaped). Input angles are $\Theta_1 = 110^\circ$, $\Theta_2 = 230^\circ$ and $\Theta_3 = 335^\circ$. Initial angles are $\theta_1 = 60^\circ$, $\theta_2 = 270^\circ$ and $\theta_3 = 290^\circ$. $\delta = 1^\circ$ and $p = 0.75$.
- Fig. 2 Cost function versus iteration number for a 3-D template (L-shaped). Input angles are $\Theta_1 = 28.5^\circ$, $\Theta_2 = 153.6^\circ$ and $\Theta_3 = 298.3^\circ$. Initial angles are $\theta_1 = 300^\circ$, $\theta_2 = 250^\circ$ and $\theta_3 = 200^\circ$. $\delta = 1^\circ$ and $p = 0.75$.
- Fig. 3 Cost function versus iteration number for a 3-D template (L-shaped) with annealing. Input angles are $\Theta_1 = 110^\circ$, $\Theta_2 = 230^\circ$ and $\Theta_3 = 335^\circ$. Initial angles are $\theta_1 = 60^\circ$, $\theta_2 = 270^\circ$ and $\theta_3 = 290^\circ$. $\delta = 1^\circ$ for first 200 iterations then $\delta = 0.2^\circ$ for the rest. $p = 0.75$.
- Fig. 4 Five 2-D templates.
- Fig. 5 Cost functions versus iteration number for five 2-D templates. Input is from template V with an angle $\Theta = 190^\circ$. Initial angles are $\theta_1 = 180^\circ$, $\theta_2 = 150^\circ$, $\theta_3 = 220^\circ$, $\theta_4 = 230^\circ$ and $\theta_5 = 265^\circ$ respectively. $\delta = 1^\circ$ and $p = 0.75$. Template V is shown in red.
- Fig. 6 Five projections of 3-D templates.
- Fig. 7 Cost function versus iteration number for resource allocation with 2 guns and 3 targets. $\delta = 0.015$, $p = 0.75$, $\sigma_{ijk} = 0$ and ϵ_{ij} as shown in Table 3(a).
- Fig. 8 Cost function versus iteration number for resource allocation with 4 guns and 4 targets. $\delta = 0.02$, $p = 0.8$, $\sigma_{ijk} = 0$ and ϵ_{ij} as shown in Table 4(a).
- Fig. 9 Cost function versus iteration number for resource allocation with 4 guns and 4 targets. $\delta = 0.01$, $p = 0.75$, $\sigma_{ijk} = 0$ and ϵ_{ij} as shown in Table 5(a).
- Fig. 10 Cost function versus iteration number for resource allocation with 4 guns and 4 targets. $\delta = 0.02$, $p = 0.8$, $\sigma_{ijk} = +0.1$ in (a), $\sigma_{ijk} = -0.1$ in (b) and ϵ_{ij} as shown in Table 5(a).

Fig. 11 A plot of CPU time for 1000 iterations versus number of parameters.

Fig. 12 Cost function versus iteration number for resource allocation with 2 guns and 3 targets. Efficiencies are given in Table 3(a). $\sigma_{ijk} = 0.0$. (a) Alopex with $\delta = 0.015$, $p = 0.78$. (b) SA with $L = 50$, $\beta = 0.5$ and $\delta = 0.015$.

Fig. 13 Cost function versus iteration number for resource allocation with 4 guns and 4 targets. Efficiencies are given in Table 6. $\sigma_{ijk} = 0.0$. (a) Alopex with $\delta = 0.015$, $p = 0.75$. (b) SA with $L = 100$, $\beta = 0.5$ and $\delta = 0.015$.

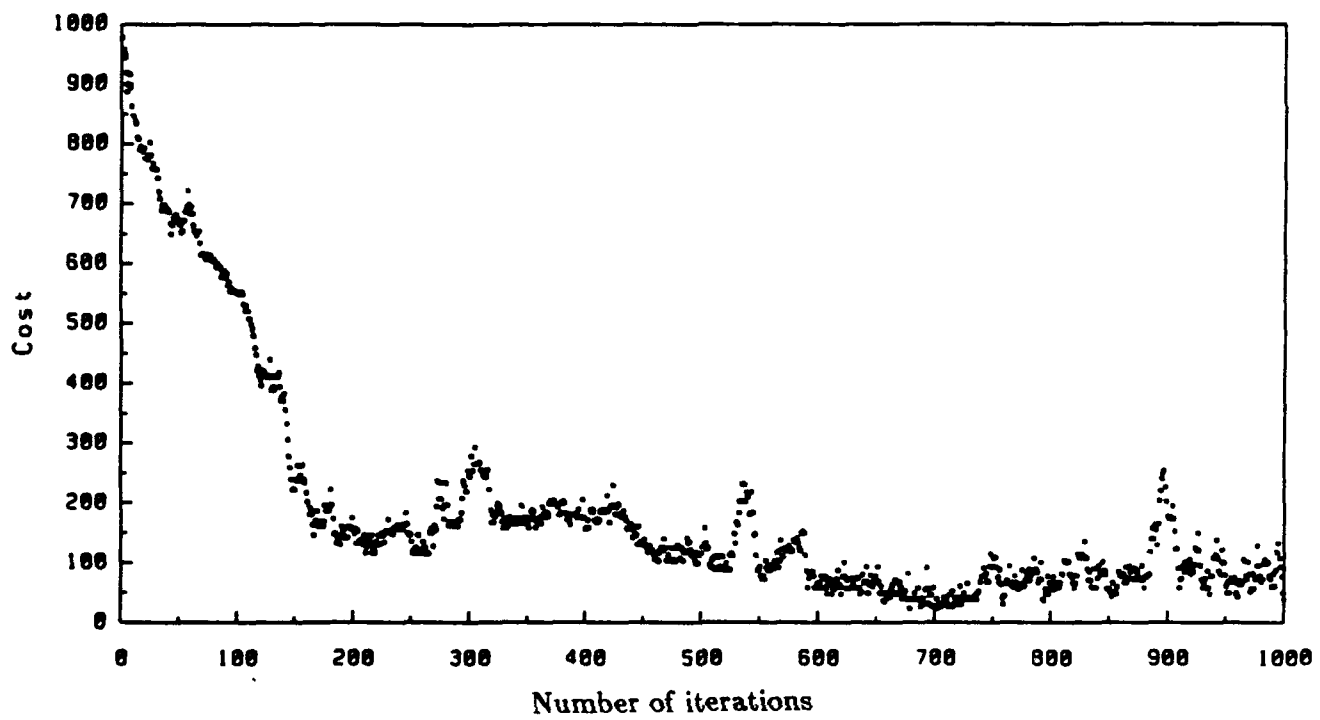


Fig. 1

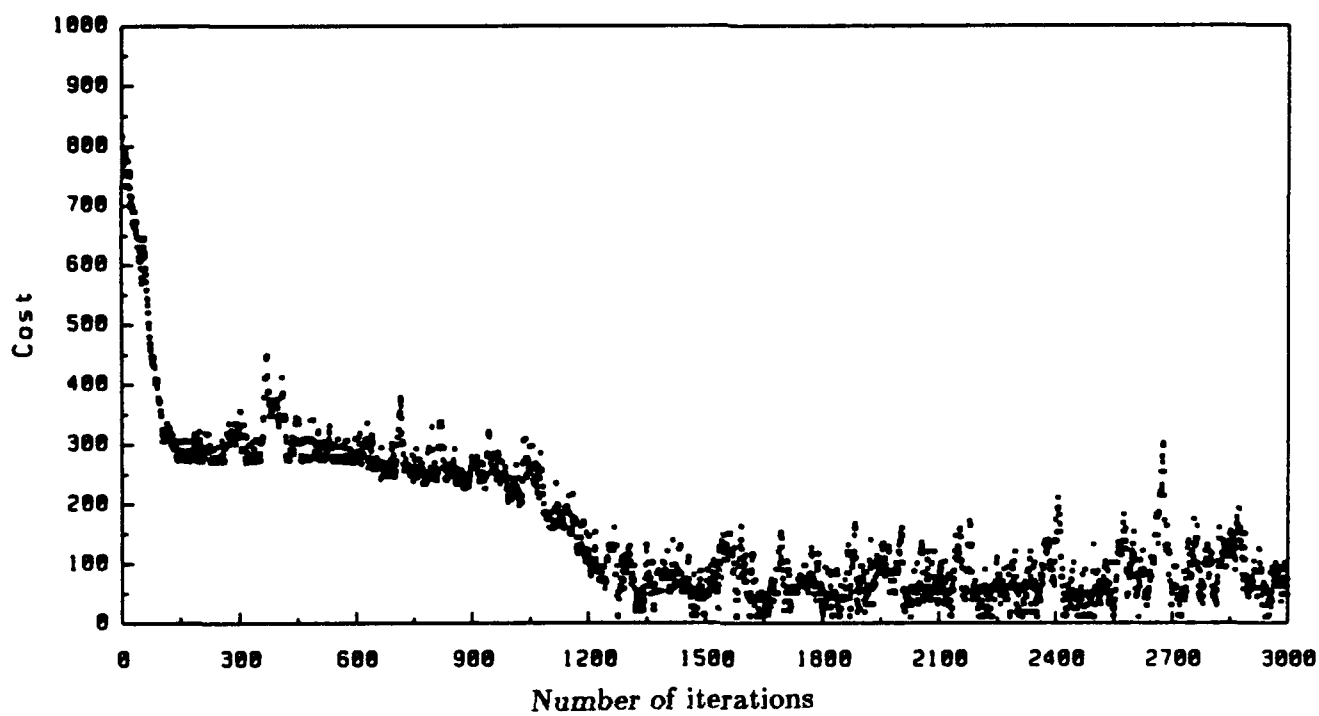


Fig. 2

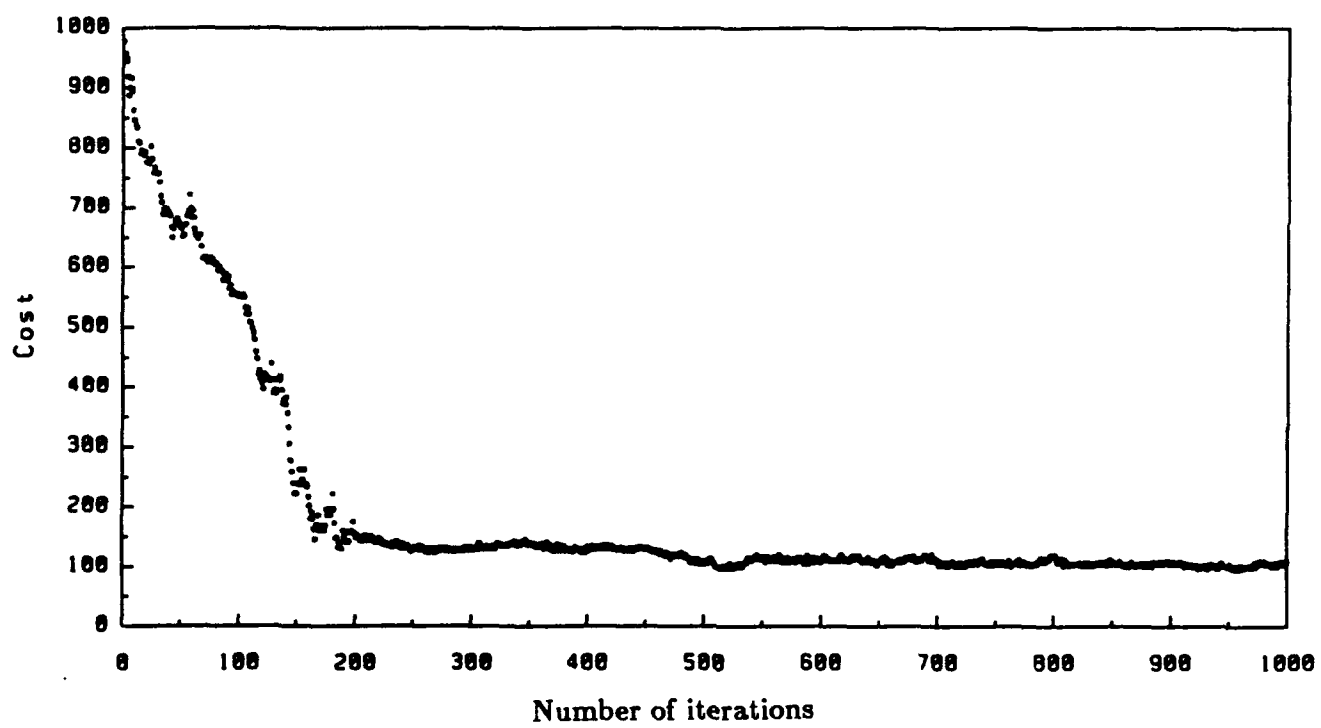


Fig. 3

Fig. 4(I)

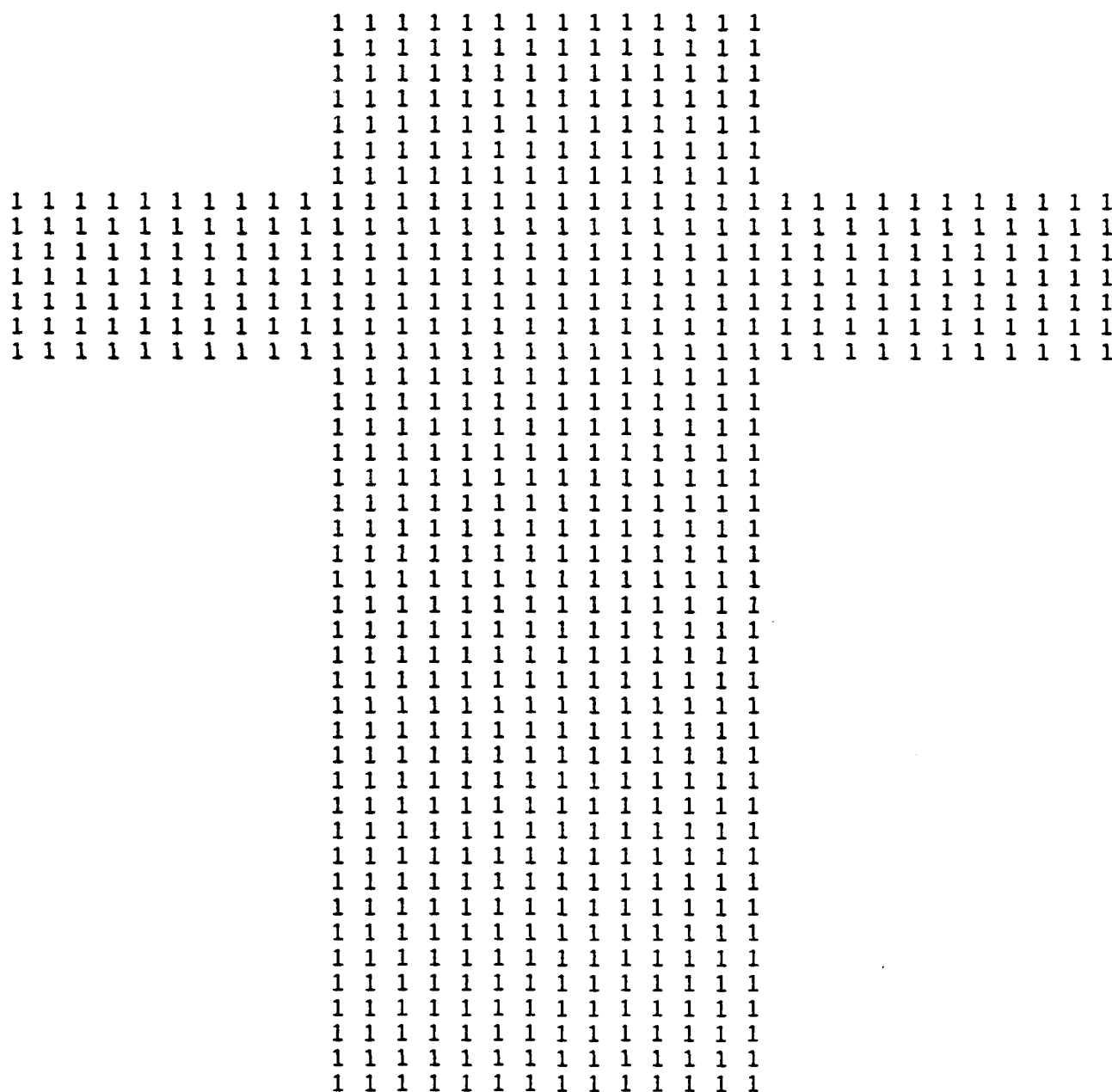


Fig. 4(II)

[illegible]

Fig. 4(III)

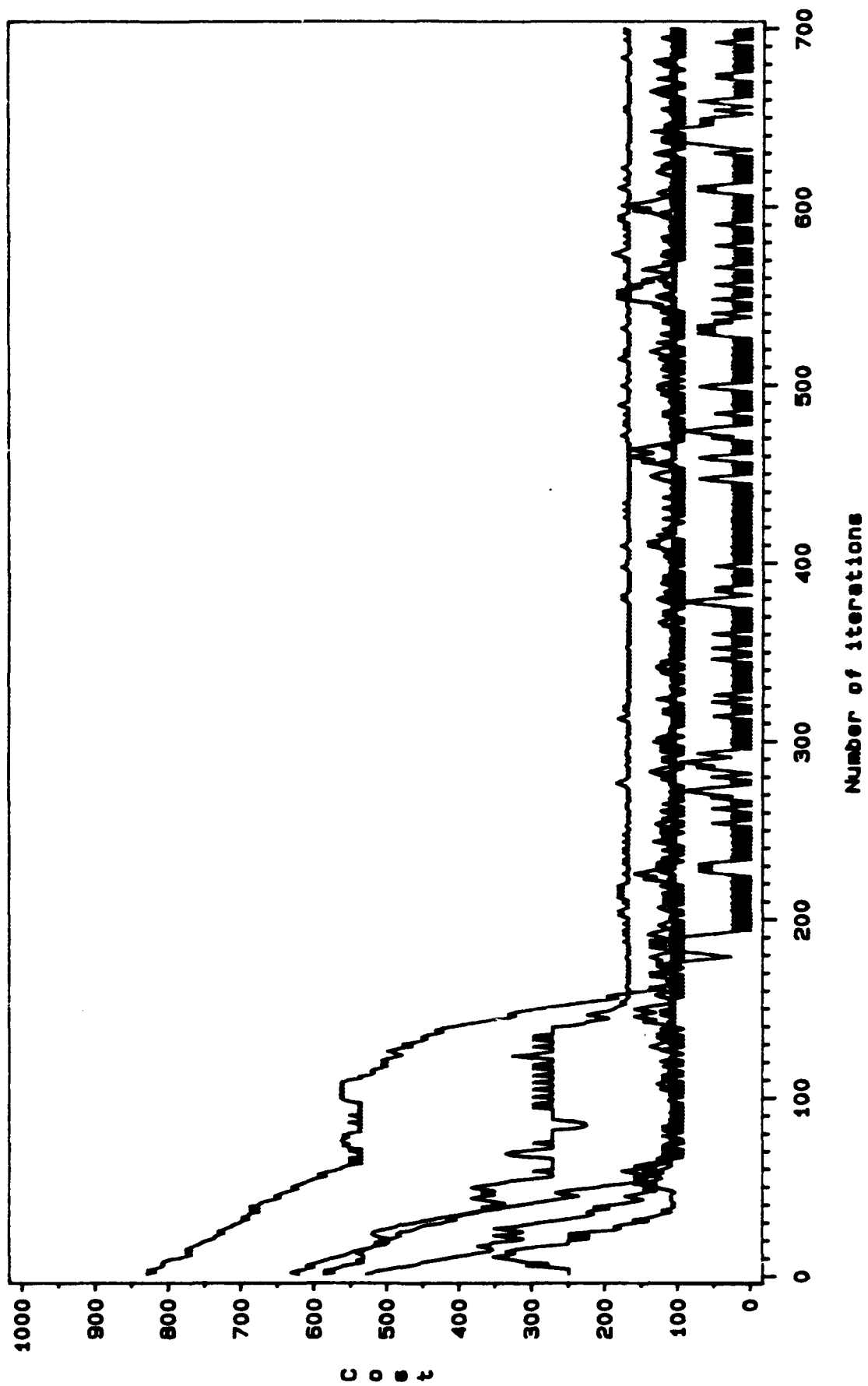


Fig. 5

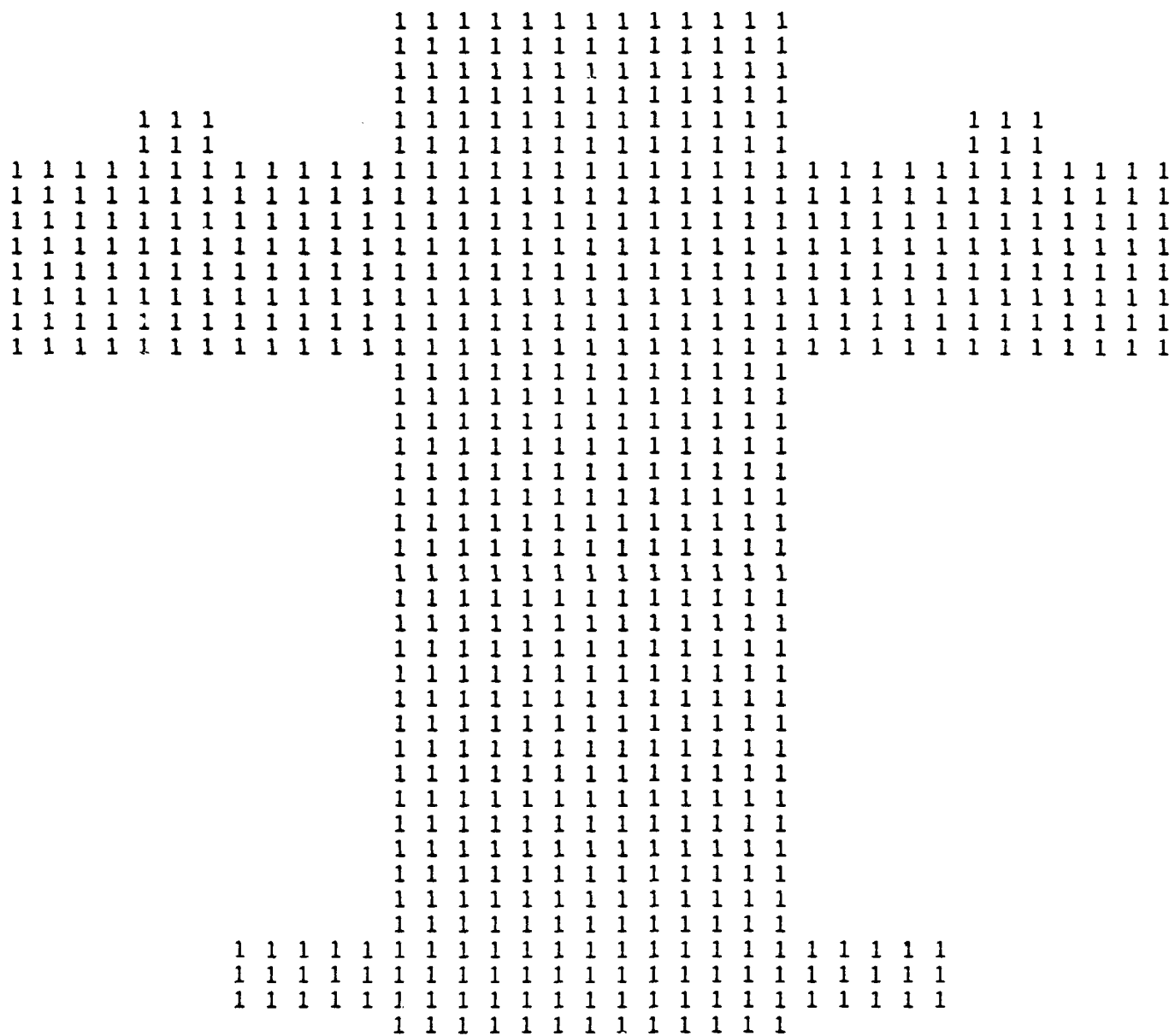


Fig. 6(II)

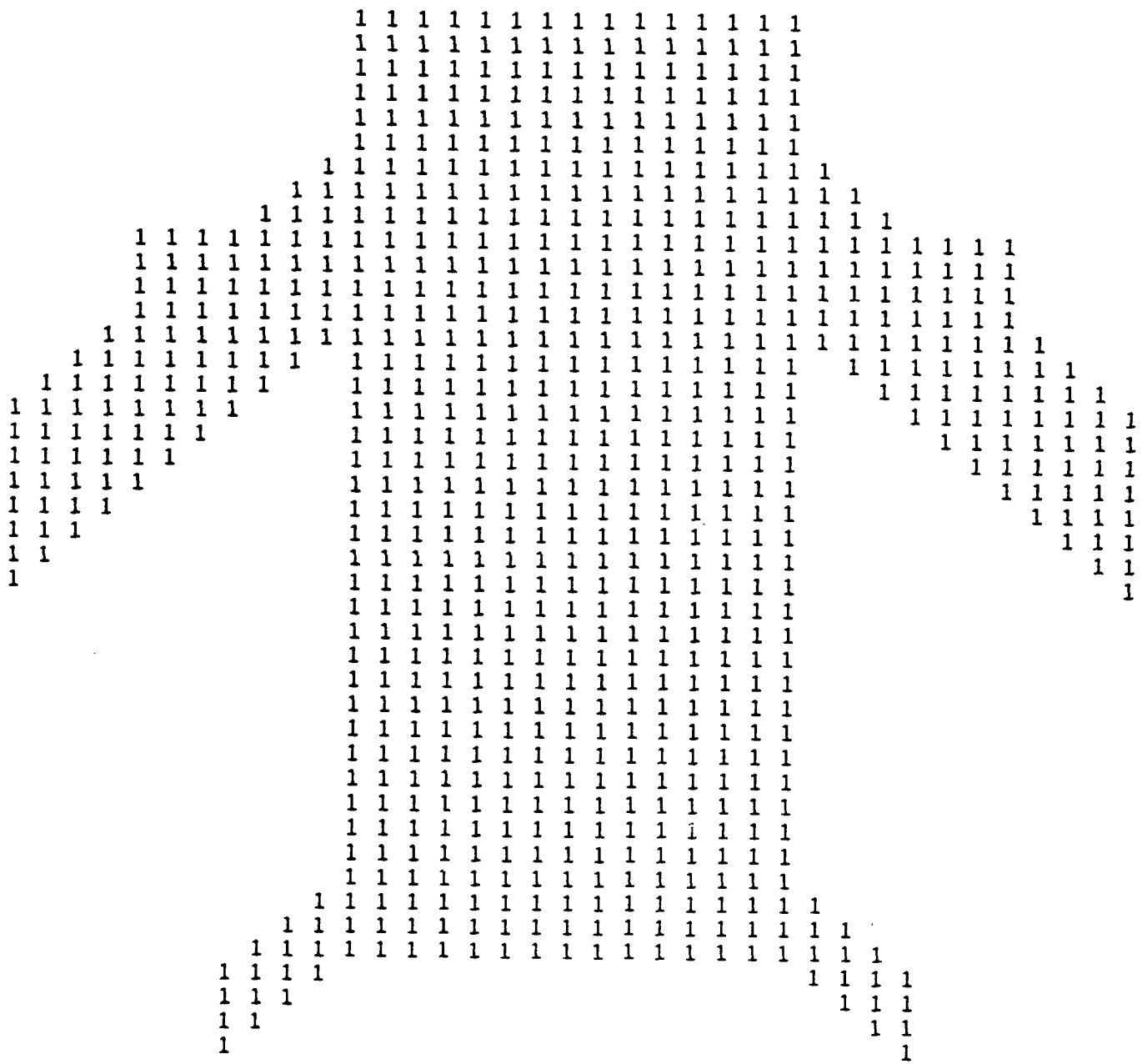


Fig. 6(IV)

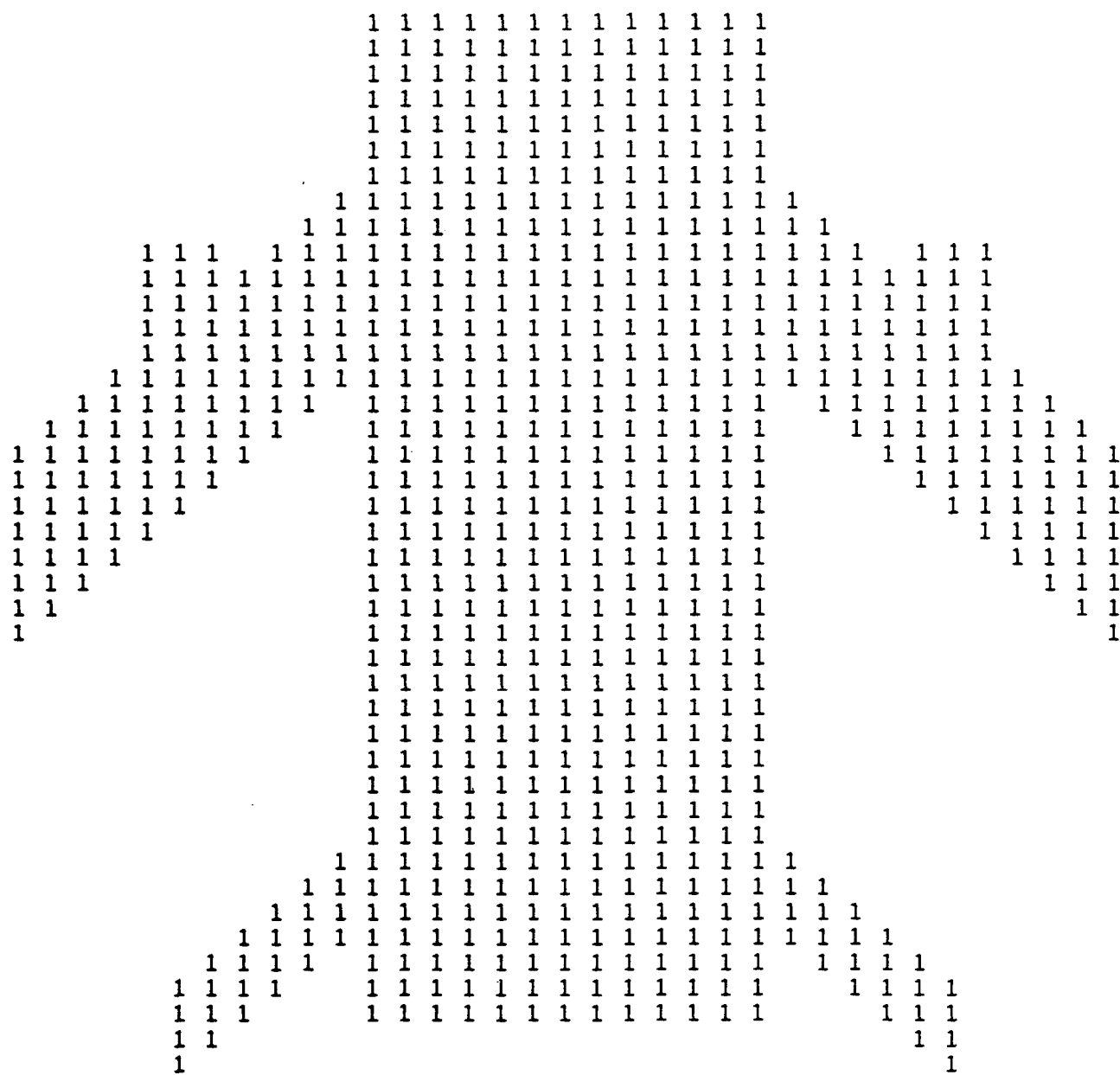


Fig. 6(V)

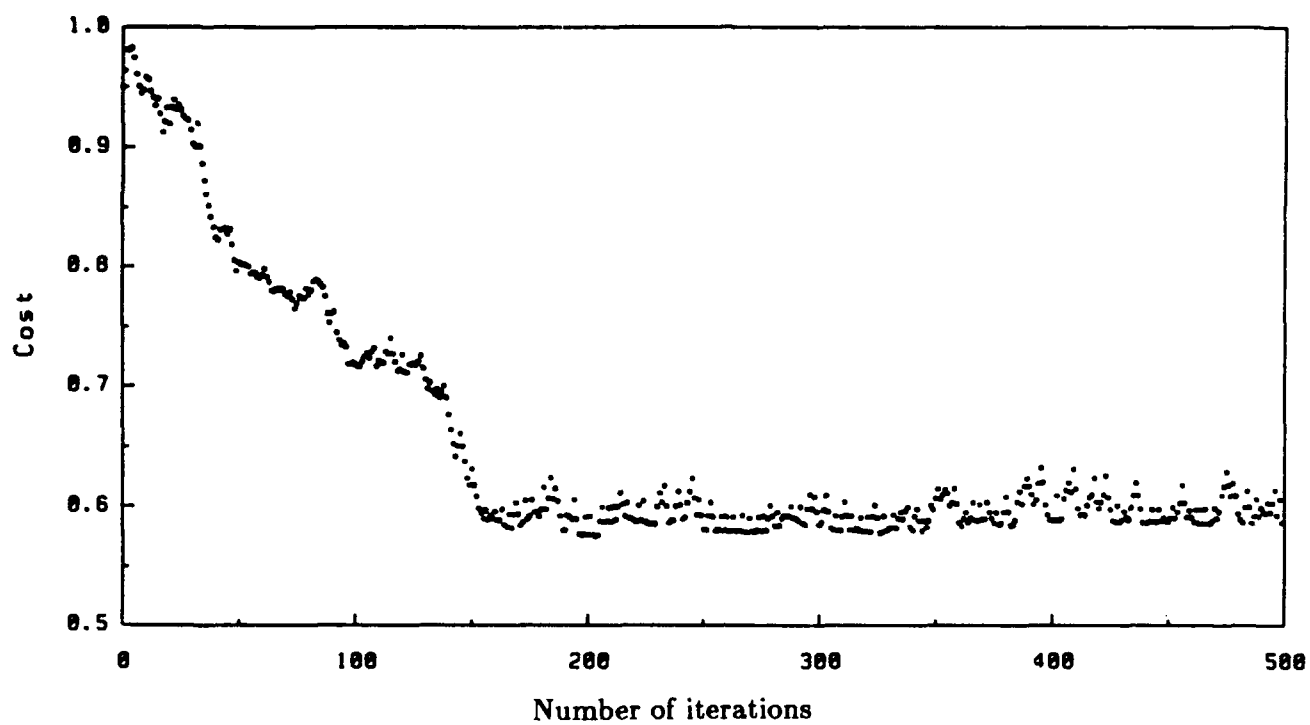
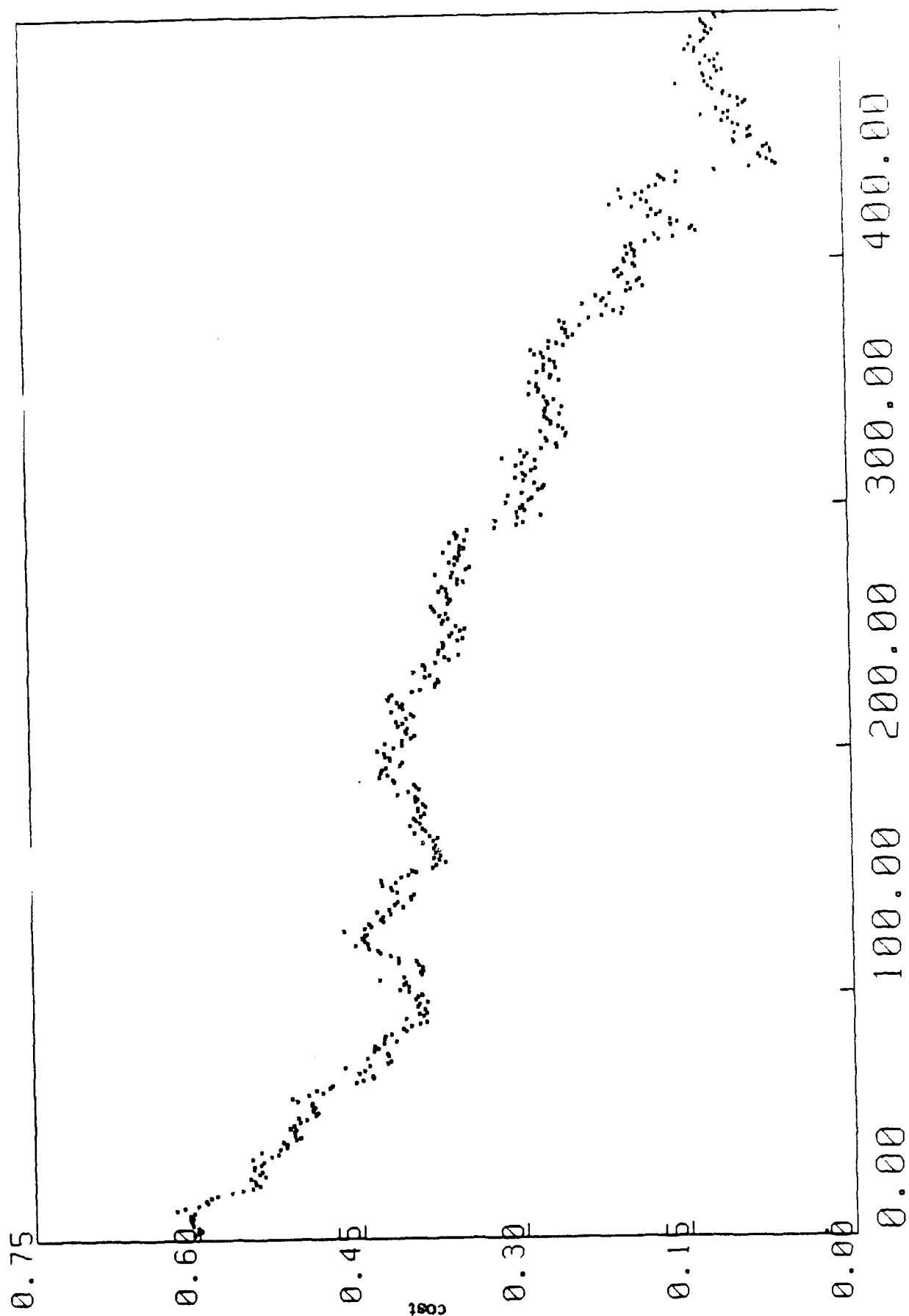


Fig. 7



Number of iterations

Fig. 8

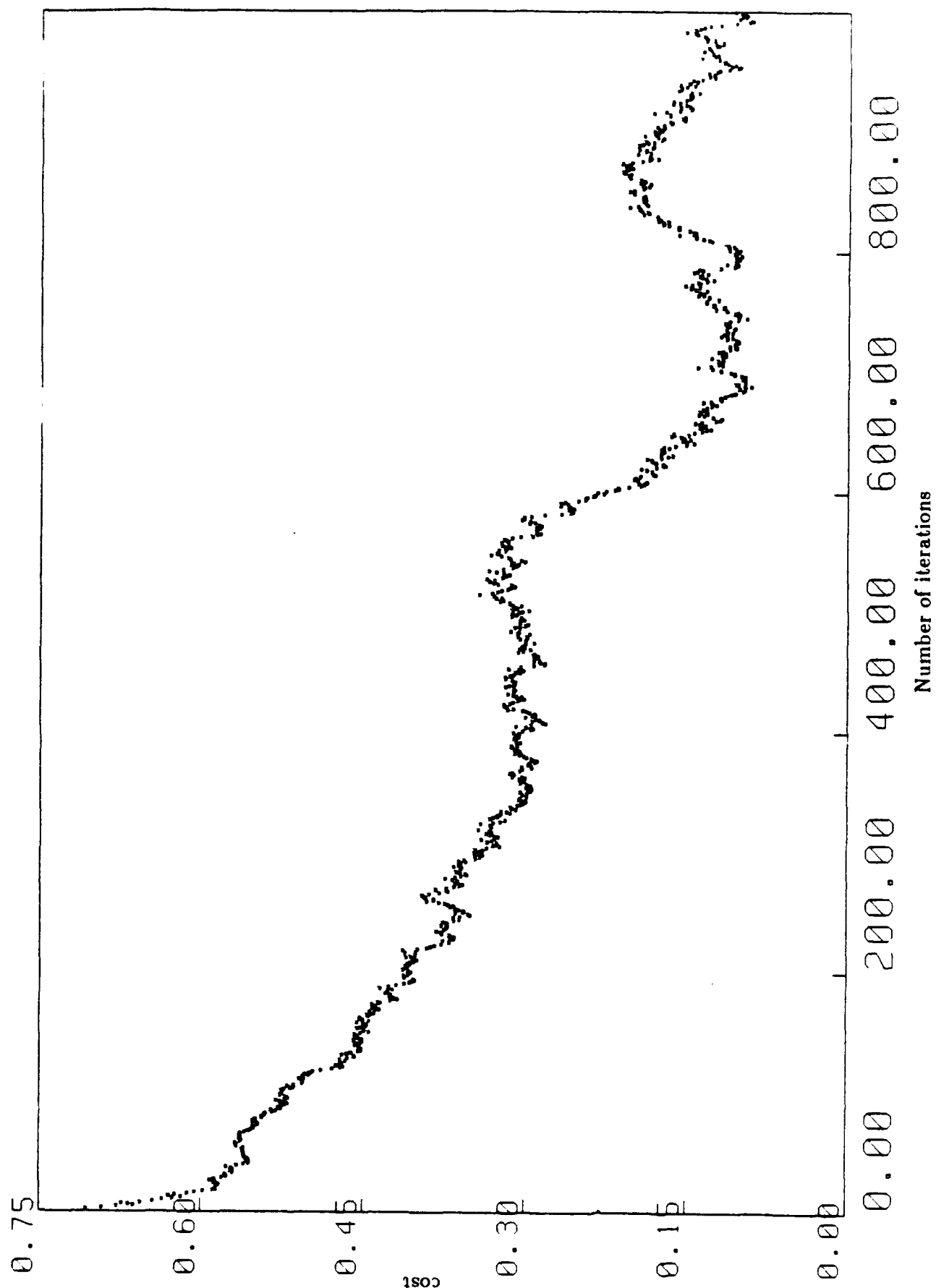
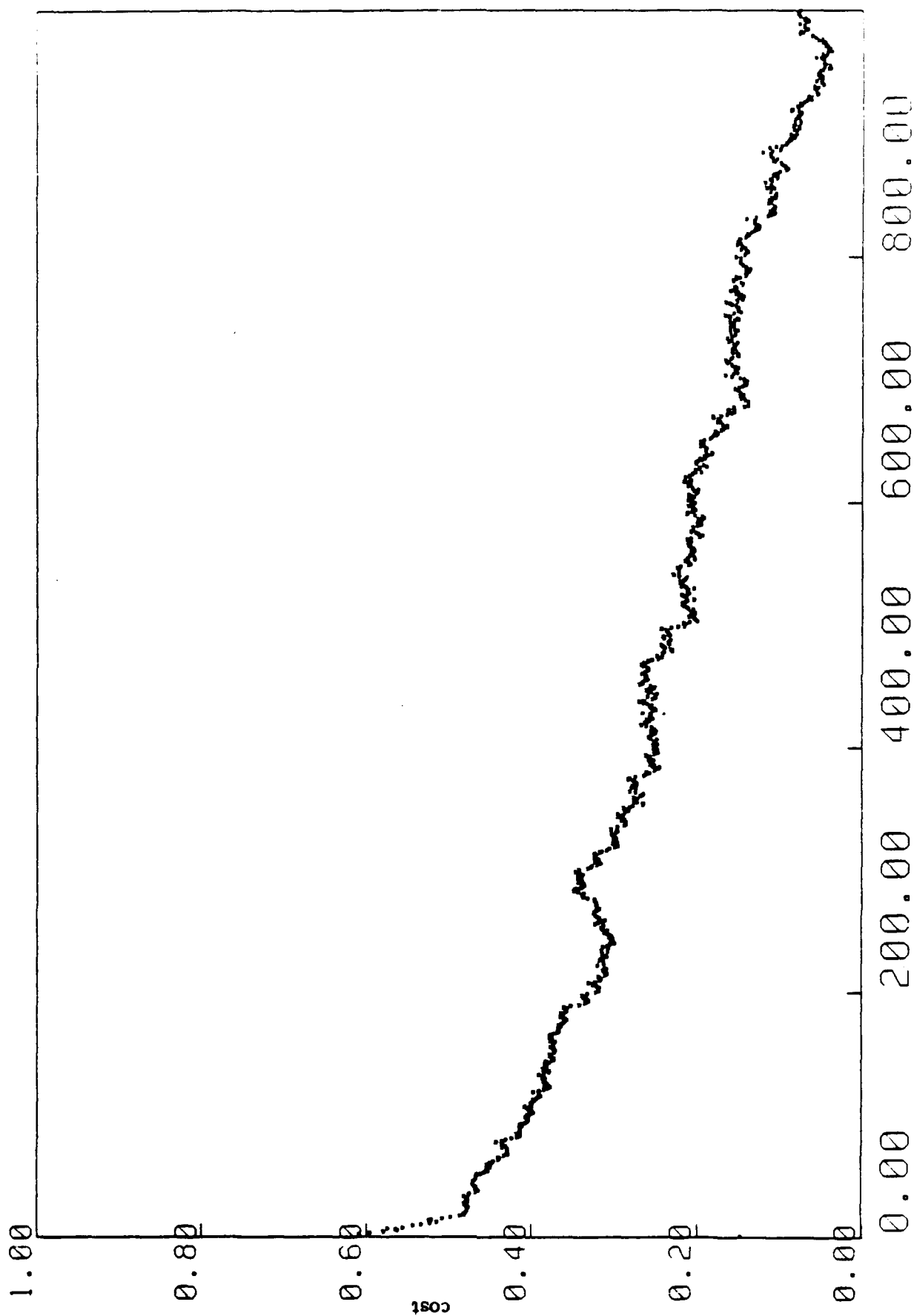
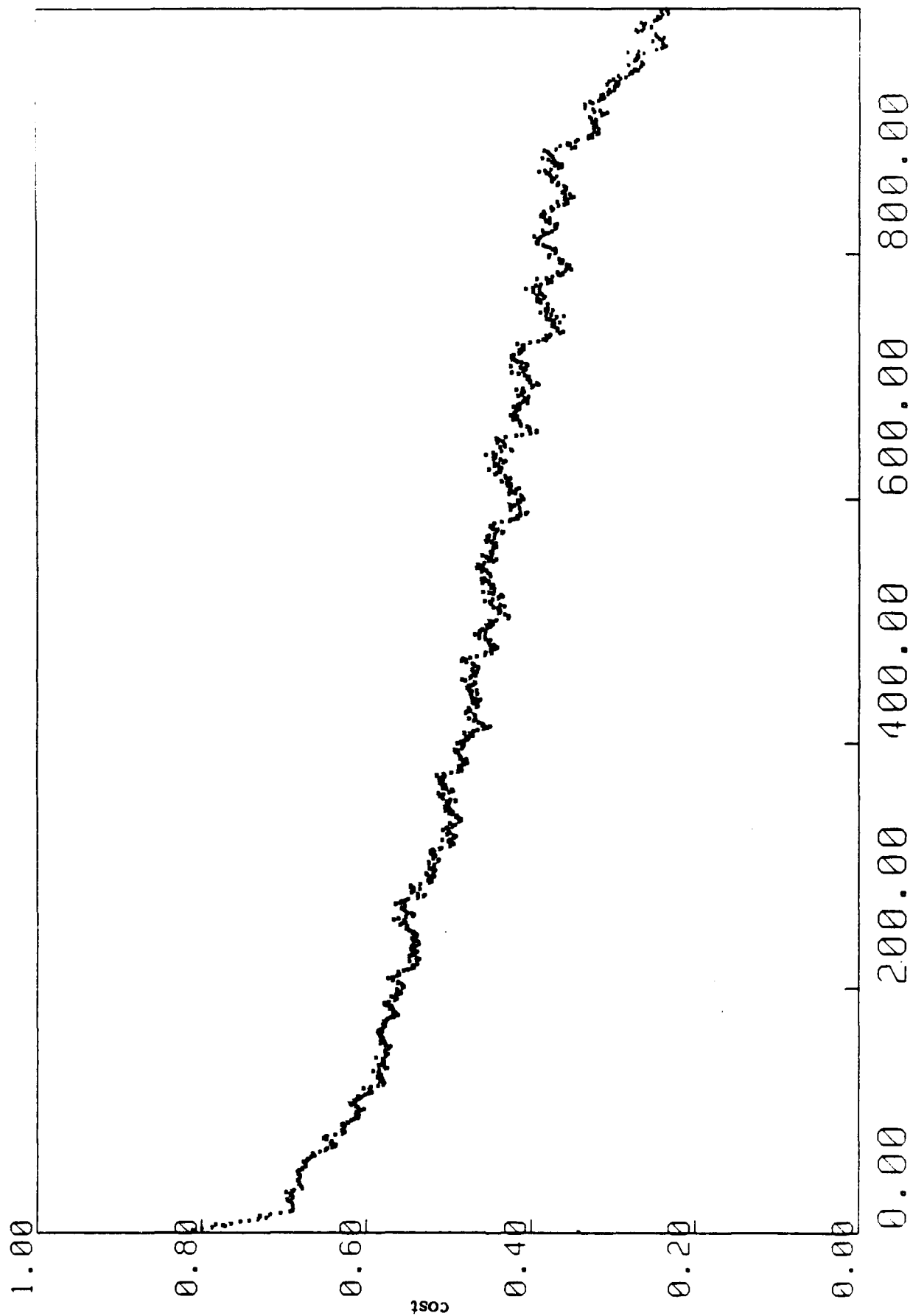


Fig. 9



Number of iterations

Fig. 10(a)



Number of iterations

Fig. 10(b)

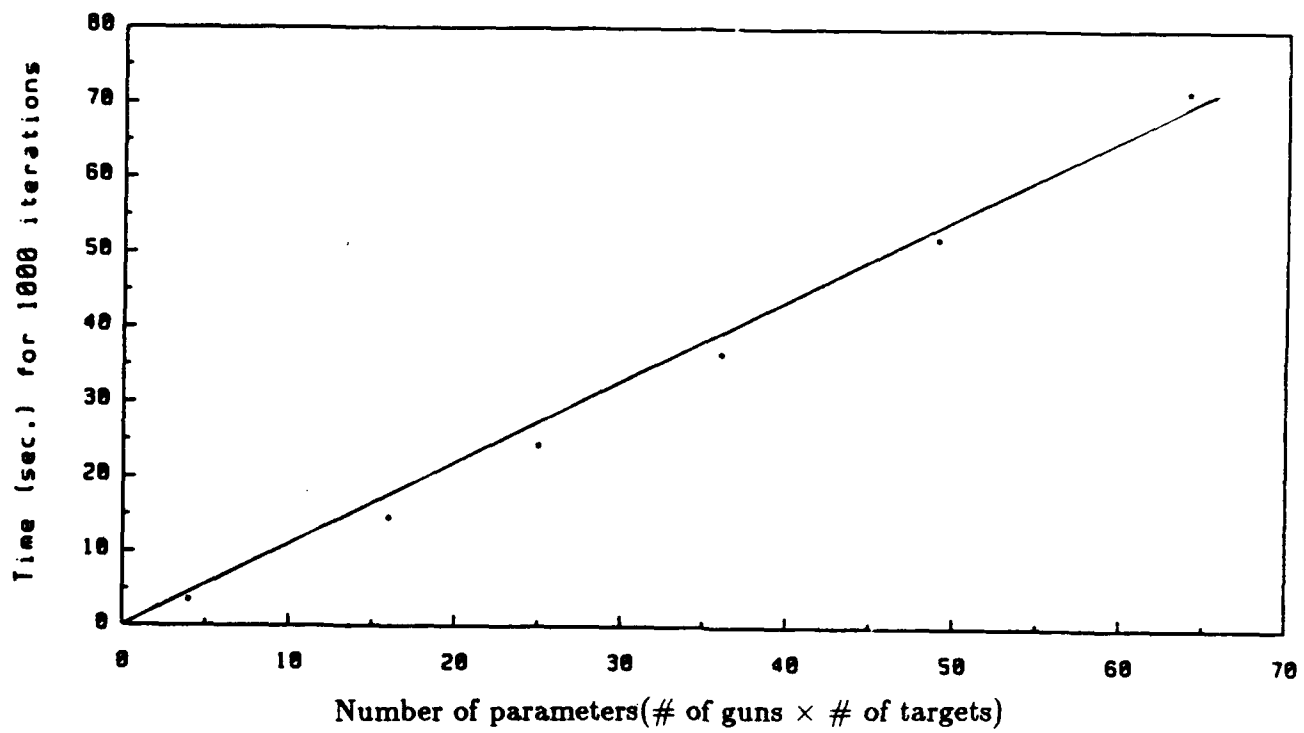


Fig. 11

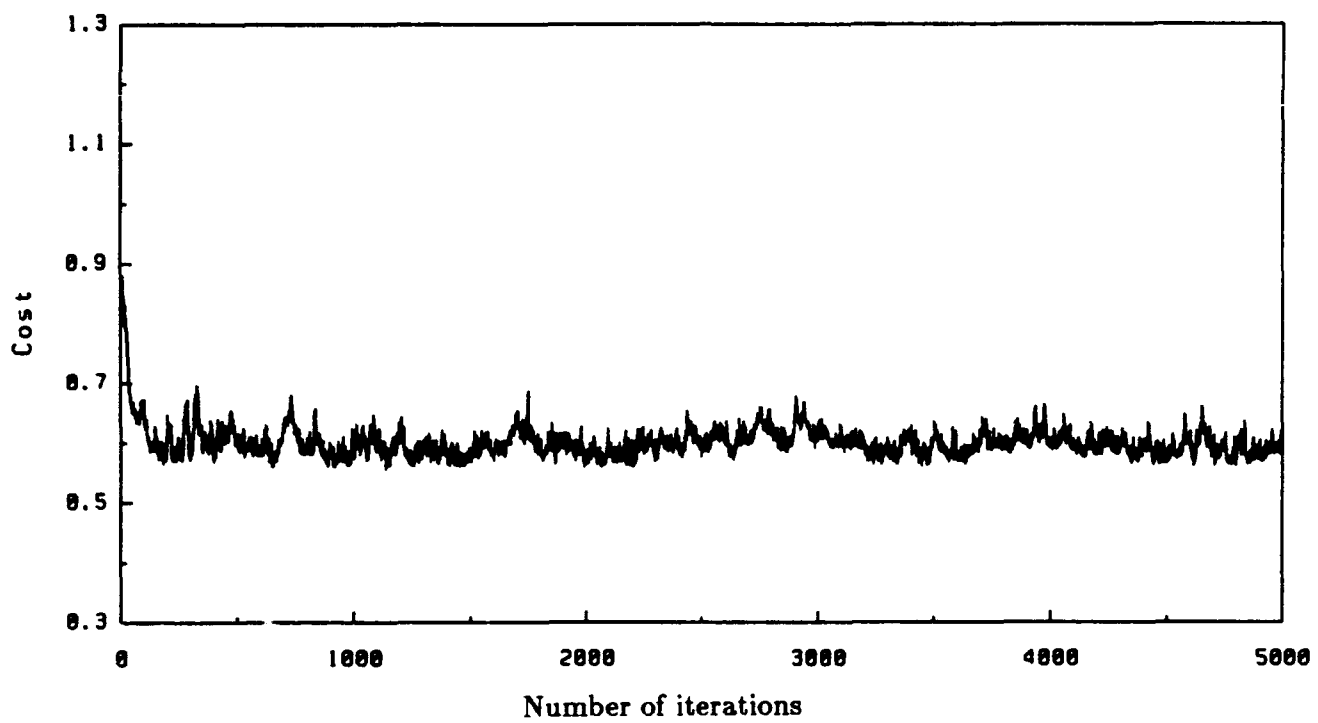


Fig. 12(a)

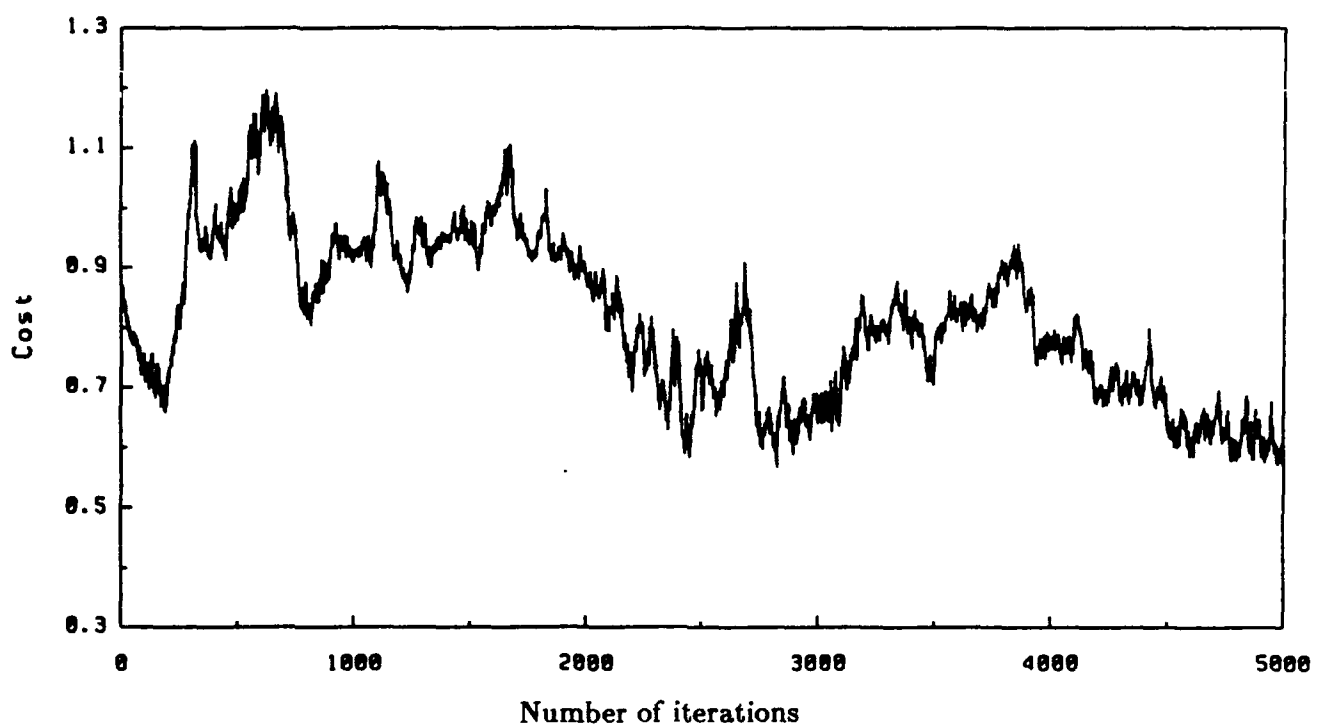


Fig. 12(b)

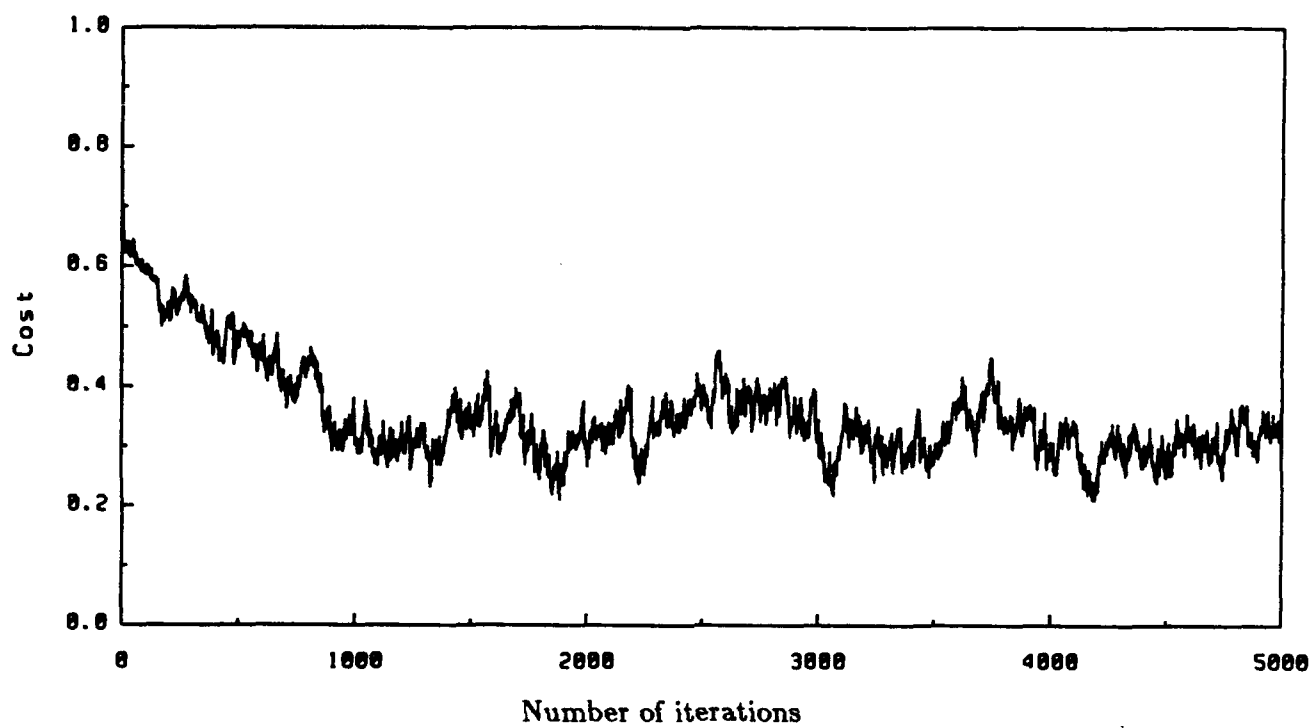


Fig. 13(a)

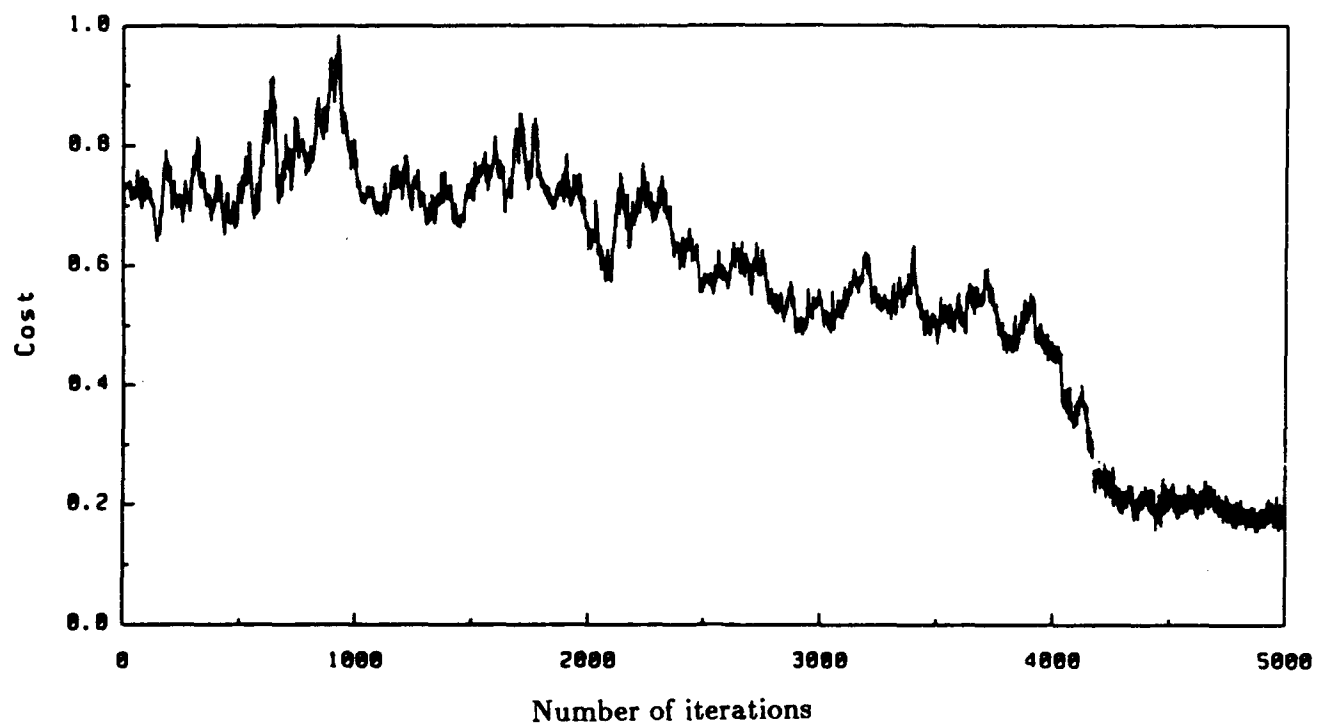


Fig. 13(b)

Table 1

Trial	1	2	3	4	5	6	Trial	1	2	3	4	5	6
F_{min}	128	80	127	79	9	24	F_{min}	38	40	85	38	66	24
Templ.	3	1	3	1	2	2	Templ.	2	2	5	2	4	2
Iter.	472	350	449	298	424	681	Iter.	382	370	483	234	589	504

(1) Input template 2, $\alpha = 80^\circ$, $\beta = 266^\circ$, $\gamma = 55^\circ$. (2) Input template 2, $\alpha = 38^\circ$, $\beta = 165^\circ$, $\gamma = 23^\circ$.

Trial	1	2	3	4	5	6	Trial	1	2	3	4	5	6
F_{min}	13	10	102	102	260	264	F_{min}	107	13	112	62	13	120
Templ.	2	2	3	3	2	1	Templ.	5	2	5	1	2	5
Iter.	491	468	351	502	665	553	Iter.	123	355	253	574	672	333

(3) Input template 2, $\alpha = 133^\circ$, $\beta = 210^\circ$, $\gamma = 75^\circ$. (4) Input template 2, $\alpha = 311^\circ$, $\beta = 322^\circ$, $\gamma = 162^\circ$.

Table 2

Trial	1	2	3	4	5	6	Trial	1	2	3	4	5	6
F_{min}	222	138	12	16	165	321	F_{min}	21	68	69	68	69	11
Templ.	1	5	4	4	5	4	Templ.	4	5	5	5	4	4
Iter.	659	441	594	170	629	659	Iter.	635	258	584	246	491	241

(1) Input template 4, $\alpha = 80^\circ$, $\beta = 266^\circ$, $\gamma = 55^\circ$. (2) Input template 4, $\alpha = 311^\circ$, $\beta = 322^\circ$, $\gamma = 162^\circ$.

Trial	1	2	3	4	5	6	Trial	1	2	3	4	5	6
F_{min}	27	70	76	12	19	19	F_{min}	262	13	7	271	9	288
Templ.	4	2	2	4	4	4	Templ.	1	4	4	2	4	5
Iter.	535	343	504	137	498	632	Iter.	601	169	285	264	549	669

(3) Input template 4, $\alpha = 38^\circ$, $\beta = 165^\circ$, $\gamma = 23^\circ$. (4) Input template 4, $\alpha = 256^\circ$, $\beta = 70^\circ$, $\gamma = 130^\circ$.

Table 3

Target		1	2	3
G u n	1	0.70	0.95	0.83
	2	0.90	0.65	0.90

(a): Efficiencies

Target		1	2	3
G u n	1	0.00	0.66	0.34
	2	0.72	0.00	0.28
Prob. of hit		0.65	0.63	0.46

(b): Assignments

Table 4

Target		1	2	3	4
G u n	1	0.70	0.95	0.85	0.91
	2	0.90	0.75	0.99	0.87
	3	0.95	0.85	0.80	0.88
	4	0.80	0.85	0.60	0.99

(a): Efficiencies

Target		1	2	3	4
G u n	1	0.00	0.94	0.04	0.01
	2	0.06	0.00	0.92	0.02
	3	0.86	0.08	0.00	0.05
	4	0.11	0.02	0.01	0.85
Prob. of hit		0.82	0.82	0.80	0.95

(b): Assignments

Table 5

Target	1	2	3	4	
G u n	1	0.90	0.95	0.75	0.95
	2	0.90	0.85	0.95	0.70
	3	0.90	0.85	0.70	0.60
	4	0.90	0.85	0.95	0.80

(a): Efficiencies

Target	1	2	3	4	
GUN	1	0.02	0.15	0.70	0.96
	2	0.97	0.00	0.02	0.00
	3	0.02	0.96	0.00	0.01
	4	0.00	0.00	0.99	0.00
Prob. of hit	0.82	0.83	0.87	0.81	

(b): Assignments

Table 6

Target		1	2	3	4
G u n	1	0.65	0.85	0.70	0.90
	2	0.80	0.80	0.50	0.85
	3	0.70	0.60	0.80	0.80
	4	0.70	0.90	0.90	0.80

Efficiencies



MISSION of Rome Air Development Center

RADC plans and executes research, development, test and selected acquisition programs in support of Command, Control, Communications and Intelligence (C³I) activities. Technical and engineering support within areas of competence is provided to ESD Program Offices (POs) and other ESD elements to perform effective acquisition of C³I systems. The areas of technical competence include communications, command and control, battle management information processing, surveillance sensors, intelligence data collection and handling, solid state sciences, electromagnetics, and propagation, and electronic reliability/maintainability and compatibility.